Supplementary Material of P-Tucker

Sejoon Oh*, Namyong Park†, Lee Sael‡, U Kang§

* *Seoul National University, Korea*

† *Carnegie Mellon University, USA*

*ohhenrie@snu.ac.kr †namyongp@cs.cmu.edu * saellee@gmail.com *ukang@snu.ac.kr

Abstract

In this supplementary material, we suggest full proof of the row-wise update rule for factor matrices and theoretical complexities of P-TUCKER-APPROX, which were introduced in the main paper.

I. FULL PROOF OF THE ROW-WISE UPDATE RULE

Definition 1 (Sparse Tucker Factorization): Given a tensor $\mathfrak{X}$ ($\in \mathbb{R}^{I_1\times\ldots\times I_N}$) with observable entries Ω, the goal of sparse Tucker factorization of $\mathfrak{X}$ is to find factor matrices $\mathbf{A}^{(n)}$ ($\in \mathbb{R}^{I_n\times J_n}$) and a core tensor $\mathcal{G}$ ($\in \mathbb{R}^{J_1\times\ldots\times J_N}$), which minimize Equation (1).

$$L(\mathcal{G},\mathbf{A}^{(1)},...,\mathbf{A}^{(N)}) = \sum_{\forall(i_1,...,i_N)\in\Omega}\left(\mathfrak{X}_{(i_1,...,i_N)} - \sum_{\forall(j_1,...,j_N)\in\mathcal{G}}\mathcal{G}_{(j_1,...,j_N)}\prod_{n=1}^{N}a^{(n)}_{i_nj_n}\right)^2 + \lambda\sum_{n=1}^{N}\|\mathbf{A}^{(n)}\|^2 \tag{1}$$

Theorem 1 (Row-wise Update rule for Factor Matrices):

$$\underset{[a^{(n)}_{i_n1},...,a^{(n)}_{i_nJ_n}]}{\arg\min}\; L(\mathcal{G},\mathbf{A}^{(1)},...,\mathbf{A}^{(N)}) = \mathbf{c}^{(n)}_{i_n:}\times[\mathbf{B}^{(n)}_{i_n}+\lambda\mathbf{I}_{J_n}]^{-1} \tag{2}$$

$$\text{where the } (j_1,j_2)\text{th entry of } \mathbf{B}^{(n)}_{i_n}(\in\mathbb{R}^{J_n\times J_n}): \sum_{\forall(i_1,...,i_N)\in\Omega^{(n)}_{i_n}}\delta^{(n)}_{(i_1,...,i_N)}(j_1)\delta^{(n)}_{(i_1,...,i_N)}(j_2), \tag{3}$$

$$\text{the } j\text{th entry of } \mathbf{c}^{(n)}_{i_n:}(\in\mathbb{R}^{J_n}): \sum_{\forall(i_1,...,i_N)\in\Omega^{(n)}_{i_n}}\mathfrak{X}_{(i_1,...,i_N)}\delta^{(n)}_{(i_1,...,i_N)}(j), \tag{4}$$

$$\text{the } j\text{th entry of } \delta^{(n)}_{(i_1,...,i_N)}(\in\mathbb{R}^{J_n}): \sum_{\forall(j_1...j_{n-1},j,j_{n+1}...j_N)\in\mathcal{G}}\mathcal{G}_{(j_1...j_{n-1},j,j_{n+1}...j_N)}\prod_{k\neq n}a^{(k)}_{i_kj_k}. \tag{5}$$

Proof:

$$\frac{\partial L}{\partial a^{(n)}_{i_nj_n}} = 0, \forall j_n, 1\le j_n\le J_n$$

$$\Leftrightarrow \sum_{\forall(i_1,...,i_N)\in\Omega^{(n)}_{i_n}}\left(\left(\mathfrak{X}_{(i_1,...,i_N)} - \sum_{\forall(j_1,...,j_N)\in\mathcal{G}}\mathcal{G}_{(j_1,...,j_N)}\prod_{n=1}^{N}a^{(n)}_{i_nj_n}\right)\times\left(-\delta^{(n)}_{(i_1,...,i_N)}(j_n)\right)\right) + \lambda a^{(n)}_{i_nj_n} = 0$$

$$\Leftrightarrow \sum_{\forall(i_1,...,i_N)\in\Omega^{(n)}_{i_n}}\left(\left(\sum_{t=1}^{J_n}\delta^{(n)}_{(i_1,...,i_N)}(t)a^{(n)}_{i_nt}\right)\times\left(\delta^{(n)}_{(i_1,...,i_N)}(j_n)\right)\right)+\lambda a^{(n)}_{i_nj_n} = \sum_{\forall(i_1,...,i_N)\in\Omega^{(n)}_{i_n}}\left(\mathfrak{X}_{(i_1,...,i_N)}\delta^{(n)}_{(i_1,...,i_N)}(j_n)\right) \tag{6}$$

$\sum_{\forall (i_1,\ldots,i_N)\in\Omega^{(n)}_{i_n}} \left(\left(\sum_{t=1}^{J_n} \delta^{(n)}_{(i_1,\ldots,i_N)}(t) a^{(n)}_{i_n t} \right) \times \left(\delta^{(n)}_{(i_1,\ldots,i_N)}(j_n) \right) \right)$ is expressed as an inner product of the following vectors.

$$\text{Row vector } (1 \times J_n) : [a^{(n)}_{i_n 1}, \ldots, a^{(n)}_{i_n J_n}]$$

$$\text{Column vector } (J_n \times 1) : \begin{bmatrix} \sum_{\forall (i_1,\ldots,i_N)\in\Omega^{(n)}_{i_n}} \delta^{(n)}_{(i_1,\ldots,i_N)}(1)\delta^{(n)}_{(i_1,\ldots,i_N)}(j_n) \\ \vdots \\ \sum_{\forall (i_1,\ldots,i_N)\in\Omega^{(n)}_{i_n}} \delta^{(n)}_{(i_1,\ldots,i_N)}(J_n)\delta^{(n)}_{(i_1,\ldots,i_N)}(j_n) \end{bmatrix}$$

If we vary j_n from 1 to J_n, the row vector is fixed as $a^{(n)}_{i_n:}$ and the column vector differs.
Thus, we can integrate each column vector into a matrix $\mathbf{B}^{(n)}_{i_n} (\in \mathbb{R}^{J_n \times J_n})$

$$\text{where the } (j_1, j_2)\text{th entry of } \mathbf{B}^{(n)}_{i_n} (\in \mathbb{R}^{J_n \times J_n}) : \sum_{\forall (i_1,\ldots,i_N)\in\Omega^{(n)}_{i_n}} \delta^{(n)}_{(i_1,\ldots,i_N)}(j_1)\delta^{(n)}_{(i_1,\ldots,i_N)}(j_2).$$

λ term is simply transformed into $\lambda \mathbf{I}_{J_n}$, where $\mathbf{I}_{J_n}$ is an identity matrix $(\in \mathbb{R}^{J_n \times J_n})$.
In the same way, the right part of Equation (6) is integrated as $\mathbf{c}^{(n)}_{i_n:} (\in \mathbb{R}^{J_n})$

$$\text{where the } j\text{th entry of } \mathbf{c}^{(n)}_{i_n:} (\in \mathbb{R}^{J_n}) : \sum_{\forall (i_1,\ldots,i_N)\in\Omega^{(n)}_{i_n}} \mathfrak{X}_{(i_1,\ldots,i_N)} \delta^{(n)}_{(i_1,\ldots,i_N)}(j).$$

Therefore, Equation (6) is equivalent to

$$[a^{(n)}_{i_n 1}, \ldots, a^{(n)}_{i_n J_n}] \times [\mathbf{B}^{(n)}_{i_n} + \lambda \mathbf{I}_{J_n}] = \mathbf{c}^{(n)}_{i_n:}$$

Since $\mathbf{B}^{(n)}_{i_n}$ is represented as the sum of rank-1 matrices and $\lambda > 0$, matrix $[\mathbf{B}^{(n)}_{i_n} + \lambda \mathbf{I}_{J_n}]$ is positive-definite and invertible. Hence,

$$\Leftrightarrow [a^{(n)}_{i_n 1}, \ldots, a^{(n)}_{i_n J_n}] = \mathbf{c}^{(n)}_{i_n:} \times [\mathbf{B}^{(n)}_{i_n} + \lambda \mathbf{I}_{J_n}]^{-1}$$

■

Algorithm 1: P-TUCKER for Sparse Tensors

Input : Tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$,
core tensor dimensionality $J_1, ..., J_N$, and
truncation rate p (P-TUCKER-APPROX only).

Output: Updated factor matrices $\mathbf{A}^{(n)} \in \mathbb{R}^{I_n \times J_n} (n = 1, ..., N)$,
and updated core tensor $\mathcal{G} \in \mathbb{R}^{J_1 \times J_2 \times \cdots \times J_N}$.

initialize factor matrices $\mathbf{A}^{(\mathbf{n})}$ $(n = 1, ..., N)$ and core tensor $\mathcal{G}$
repeat
  update factor matrices $\mathbf{A}^{(n)}$ $(n = 1, ..., N)$
  calculate reconstruction error
  if P-TUCKER-APPROX **then** ▷ $\mathcal{G}$ Truncation
    remove "noisy" entries of $\mathcal{G}$ by Algorithm 2
until *the maximum iteration or* $\|\mathcal{X} - \mathcal{X}'\|$ *converges*;
for n = *1...N* **do**
  $\mathbf{A}^{(n)} \rightarrow \mathbf{Q}^{(n)}\mathbf{R}^{(n)}$ ▷ QR decomposition
  $\mathbf{A}^{(n)} \leftarrow \mathbf{Q}^{(n)}$ ▷ Orthogonalize $\mathbf{A}^{(n)}$
  $\mathcal{G} \leftarrow \mathcal{G} \times_n \mathbf{R}^{(n)}$ ▷ Update core tensor $\mathcal{G}$

Algorithm 2: P-TUCKER-APPROX

Input : Tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$,
factor matrices $\mathbf{A}^{(n)} \in \mathbb{R}^{I_n \times J_n} (n = 1, ..., N)$,
core tensor $\mathcal{G} \in \mathbb{R}^{J_1 \times J_2 \times \cdots \times J_N}$, and
truncation rate p $(0 < p < 1)$.

Output: Truncated core tensor $\mathcal{G}' \in \mathbb{R}^{J_1 \times J_2 \times \cdots \times J_N}$.

for $\beta = \forall (j_1, ..., j_N) \in \mathcal{G}$ **do**
  compute a partial reconstruction error $\mathcal{R}(\beta)$
sort $\mathcal{R}(\beta)$ in a descending order with their indices
remove $p|\mathcal{G}|$ entries in $\mathcal{G}$, whose $\mathcal{R}(\beta)$ value are ranked within top-$p|\mathcal{G}|$ among all $\mathcal{R}(\beta)$ values.

II. THEORETICAL COMPLEXITIES OF P-TUCKER-APPROX

Theorem 2 (Time complexity of P-TUCKER-APPROX*):* The time complexity of P-TUCKER-APPROX is $O(NIJ^3 + N^2|\Omega||\mathcal{G}|)$.

Proof: The only difference between P-TUCKER and P-TUCKER-APPROX is that P-TUCKER-APPROX exploits $|\mathcal{G}|$ entries rather than using full J^N entries of $\mathcal{G}$. Thus, the time complexity of P-TUCKER-APPROX for updating factor matrices and computing the reconstruction error is reduced to $O(NIJ^3 + N^2|\Omega||\mathcal{G}|)$. Moreover, the cost of Algorithm 2 is $O(N|\Omega||\mathcal{G}|)$, which is much less than that of other parts. Hence, the time complexity of P-TUCKER-APPROX is $O(NIJ^3 + N^2|\Omega||\mathcal{G}|)$. ∎

Theorem 3 (Memory complexity of P-TUCKER-APPROX*):* The memory complexity of P-TUCKER-APPROX is $O(J^N)$.

Proof: Compared to P-TUCKER, P-TUCKER-APPROX requires additional intermediate data to store $\mathcal{R}(\beta)$. The memory complexity of $\mathcal{R}(\beta)$ is at most $O(J^N)$, and the memory requirements for $\mathcal{R}(\beta)$ is much larger than other intermediate data. Therefore, the memory complexity of P-TUCKER-APPROX is $O(J^N)$. ∎